# 強化学習を用いた回転型倒立振子の振り上げ制御

M2013SC001 天野敦
指導教員：大石泰章

## 1　はじめに

強化学習とは，動的システムに対して最適な入力を決定する機械学習の一種である．動的システムの状態遷移に対して報酬を与え，報酬を最大化するような入力を決定する．報酬は，制御の目標を数値化して設定する．ただし，目前の報酬を最大化する入力を決定するのではなく，その後与えられるであろう報酬の総和である価値関数を最大化するように入力を決定する．価値関数は複数の未知パラメータを使って表現し，未知パラメータを適切に更新することで学習を行う．そして，動的システムを何度も動かし，未知パラメータを最適なパラメータに近づけることで，制御問題の解決を行う [1].

強化学習を機械システムに適用するにあたり，様々な学習の方法が提案されてきた [2][3]．強化学習を適用するためには，その制御対象の特性に合った制御器の設計が必要であることが分かっている．Doya[4] は，連続な状態や入力をもつシステムに対して，強化学習の理論を提案している．しかし，制御対象の次元が増加すると共に，学習する未知パラメータの数が著しく増加するという問題がある．その結果，計算量が増加し，学習にかかる時間も増加する．

本研究では，回転型倒立振子に対して次元を限定し強化学習を適用し，制御問題の解決を行う．また，次元の増加に対する学習性能の変化について考察する．

## 2　回転型倒立振子のモデル

### 2.1　制御対象の特性と数学モデル

本研究で用いる制御対象を図 1 に示す．リンク 1 はモータが取り付けられた能動的な関節であり，入力電圧 $v(t)$ を加えることでモータを駆動し，リンク 1 のトルク $\tau_1(t)$ を決定できる．また，リンク 2 の関節はモータが取り付けられていない受動的な関節である．リンク 1 を XY 平面上で回転させることにより，リンク 1 の先端に取り付けられているリンク 2 を動かすトルク $\tau_2(t)$ を間接的に加えることができる．ただし，入力電圧 $v(t)$ は制限があり，$v^{\max}$ を超えない範囲で入力を決定する．

本研究での制御目的は，回転型倒立振子リンク 1 にトルクを加えることで，これに接続されたリンク 2 も振り上げ，倒立状態で安定化するような制御器設計とする．

ある方向を基準としたリンク 1 の角度 $\theta_1(t)$[rad] と角速度 $\dot{\theta}_1(t)$[rad/s] および鉛直方向を基準としたリンク 2 の角度 $\theta_2(t)$[rad] と角速度 $\dot{\theta}_2(t)$[rad/s] を観測状態とし，これらの値をもとに入力電圧 $v(t)$ を計算する．ラグランジュの運動方程式により制御対象の数学モデルは以下の式のようになる [6]：

$$\ddot{\theta}_1(t) = -a\dot{\theta}_1(t) - a_{\mathrm{sgn}}\mathrm{sgn}\dot{\theta}_1(t) + bv(t), \qquad (1)$$

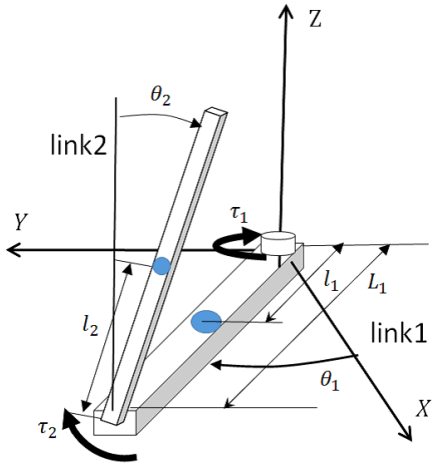


図 1　回転型倒立振子のモデル図

$$\overline{J}_2\ddot{\theta}_2(t) = \tau_2(t) + m_2gl_2\sin\theta_2(t) - c_2\dot{\theta}_2(t). \qquad (2)$$

ただし，式 (1) はリンク 1，式 (2) はリンク 2 の運動方程式を表す．また，$\tau_2(t)$ は以下の式で決まる：

$$\tau_2(t) = \overline{J}_2\dot{\theta}_1(t)^2\sin\theta_2(t)\cos\theta_2(t) - m_2L_1l_2\cos\theta_2(t)\ddot{\theta}_1(t). \qquad (3)$$

以上の式に関する物理パラメータの説明を表 1 に示す．

表 1　回転型倒立振子の物理パラメータ

| | |
|---|---|
| $a$,$a_{\mathrm{sgn}}$, $b$ | モータ，リンク 1 の特性により決まる定数 |
| $L_1$ | リンク 1 の軸から先端までの長さ |
| $m_2$ | リンク 2 の質量 |
| $l_2$ | リンク 2 の軸から重心までの長さ |
| $J_2$ | リンク 2 の重心周りの慣性モーメント |
| $c_2$ | 振子の粘性摩擦係数 |
| $g$ | 重力加速度 |

## 3　回転型倒立振子の制御

### 3.1　強化学習の適用

回転型倒立振子の制御をするために文献 [4] に習って強化学習アルゴリズムを設計する．まず，観測する状態変数は

$$x(t) = \begin{bmatrix} \dot{\theta}_1(t) & \theta_2(t) & \dot{\theta}_2(t) \end{bmatrix}^{\mathrm{T}}$$

である．式 (1)–式 (3) の微分方程式をまとめて以下のように書く：

$$\dot{x}(t) = f(x(t), v(t)).$$

報酬関数は以下の式とする：

$$r(x, v) = R(x) - S(v). \qquad (4)$$

$R(x)$ は状態変数に対する報酬関数であり，$S(v)$ は，入力制限 $|v| \leq v^{\max}$ を補償するための費用関数で次のように定める：

$$S(v) = c \int_0^v s^{-1}\left(\frac{v}{v^{\max}}\right) \mathrm{d}v. \tag{5}$$

ただし，$s(x)$ は $\frac{2}{\pi}\arctan(\frac{\pi}{2}x)$ とする．式 (5) は，$v$ が $v^{\max}$ に近づくほど，$S(v)$ は大きくなり，式 (4) で定めた報酬は少なくなる．$c$ は，入力の重みパラメータであり，$c$ を 0 の近くに選ぶほど $v^{\max}$ の範囲内で自由に入力を決定できることを意味する．

また，制御目的を達成するための制御入力として政策 $\mu$ を決める：

$$v(t) = \mu(x(t)). \tag{6}$$

### 3.2 価値関数の更新

強化学習では，学習を効率的に行うために最適な入力を決定するのだが，その決定する指標として価値関数がある．目前の報酬を取るのではなく，その後得られる報酬の累積和を最大化するような制御器の学習を行う．価値関数は以下のように定義する：

$$V^{\mu}(x(t)) = \int_t^{\infty} \mathrm{e}^{-\frac{s-t}{\tau}} r(x(s), \mu(x(s)))\, \mathrm{d}s. \tag{7}$$

ただし，$\tau > 0$ はその後得られる報酬を割り引く時定数を表す．

価値関数を式 (7) に基づいて求めることは困難であり，現実的でない．そこで，学習パラメータ $\boldsymbol{w}$ を持つ関数近似器 (Normalized Gaussian Network) によって近似する：

$$V^{\mu}(x(t)) = V(\boldsymbol{x}; \boldsymbol{w}). \tag{8}$$

式（7）と関数近似した価値関数の誤差を評価するための TD 誤差を定義する：

$$\delta(t) := r(x(t), \mu(x(t))) - \frac{1}{\tau} V^{\mu}(t) + \dot{V}^{\mu}(t). \tag{9}$$

関数 $V^{\mu}(t)$ が式 (7) の形で書けるとき式 (9) の $\delta(t)$ は零になる．TD 誤差をなくすために Eligibility Trace を用いて学習パラメータ $\boldsymbol{w}$ を更新する．TD 誤差はそれまでの時間の近似誤差を伝播しているものと考え，その伝播が式 (7) に対応して，指数関数的に増加していることと仮定する．そのとき，学習パラメータ $\boldsymbol{w}$ の更新アルゴリズムを

$$\dot{w}_k = \alpha \delta(t) e_k(t), \tag{10}$$

$$\kappa \dot{e}_k(t) = -e_k(t) + \frac{\partial V(x(t); w)}{\partial w_k} \tag{11}$$

とする．ただし，$\alpha$ は学習率であり，$\kappa$ は Eligibility Trace の時定数で，$0 < \kappa < \tau$ を満たすように設定する．

### 3.3 Normalized Gaussian Network を用いた関数近似

回転型倒立振子の状態変数の空間に Gauss 関数を配置する．Gauss 分布は以下の式となる：

$$a_k(x) = e^{-s_k \|x - c_k\|^2}. \tag{12}$$

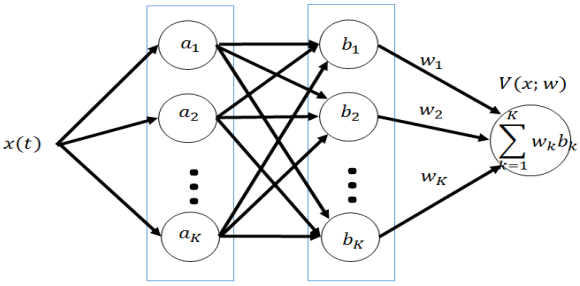


図 2 Normalized Gaussian Network

ただし，$c_k$ は Gauss 関数の中心であり，$c_1, c_2, \ldots, c_K$ を状態空間中に適切に配置する．$s_k$ は $k$ 番目の Gauss 関数の分散の設定に用いる．また，配置したすべての Gauss 関数 $a_k(x)$ について正規化した関数

$$b_k(x) = \frac{a_k(x)}{\sum_{k=1}^{K} a_k(x)} \tag{13}$$

を求める．価値関数は正規化した Gauss 関数にの線形和で近似できるとし，

$$V(\boldsymbol{x}; \boldsymbol{w}) = \sum_{k=1}^{K} w_k b_k(x) \tag{14}$$

とする．正規化された Gauss 関数の重み $w_1, w_2, \ldots, w_K$ を学習するべきパラメータとして更新していくことで価値関数を更新する．式 (12)–式 (14) までの関係を図 2 で表す．

### 3.4 入力電圧の決定則

入力電圧の決定則を考える．式 (7) の価値関数が与えられているとき，最適な入力 $v$ を求める式を以下で表す：

$$v = v^{\max} s\left(\frac{1}{c} \frac{\partial f(x,v)}{\partial v}^{\mathrm{T}} \frac{\partial V(x)}{\partial x}^{\mathrm{T}}\right). \tag{15}$$

また，本研究では制御対象の数学モデルの入力 $v$ に関する偏微分は

$$\frac{\partial f(x,v)}{\partial v} = \begin{bmatrix} b & 0 & \frac{bm_1 L_2 l_2 \cos\theta_2}{\bar{J}_2} \end{bmatrix}^{\mathrm{T}}$$

である．

## 4 シミュレーション結果

### 4.1 リンク 2 の振り上げ制御（状態空間 2 次元）

はじめに，リンク 1 を無視して，リンク 2 のみに強化学習を適用し，学習が成功するかどうかを検証する．状態変数は

$$x(t) = \begin{bmatrix} \theta_2(t) & \dot{\theta}_2(t) \end{bmatrix}^{\mathrm{T}}$$

で 2 次元となり，この空間に基底関数を配置し，価値関数を近似する．リンク 2 に加えるトルクが直接入力できると考えて $\tau_2(t) = \mu(x(t))$ とし，$\tau_2$ には制限があると考え，$\tau_2^{\max}$ を設定し強化学習を適用した．基底関数は

$[-\pi, \pi] \times \left[-\frac{5}{4}\pi, \frac{5}{4}\pi\right]$ の空間に各軸を 16 分割して格子を作り，できた格子点をそれぞれ中心として $16^2$ 個の基底関数を用意する．

本研究における試行とは，初期値から 20[s] 学習を行いながらリンク 2 を動かすことを言う．初期値は，角度は $\theta_2(0)$ をランダムで与え，角速度 $\dot{\theta}_2(0) = 0$ とする．学習を行うサンプリング時間は 0.02[s] とする．また，リンク 2 が回転しすぎることによる報酬の獲得を防ぐために $|\theta_2| > 5\pi$[rad] となったとき，報酬ではなく罰として $r(x, \tau_2) = -1$ を 1 秒間与える．また，15[s] から 20[s] までの間 $|\theta_2| < 0.2$[rad] を保ったとき，強化学習による振り上げ安定化が成功したと見なす．

図 3 は 10 回振り上げ成功後で試行 100 回目の価値関数のグラフであり．図 4 は，そのときのリンク 2 の動きを表している．このときは初期値を $\theta_2(0) = -2.14$ としてシミュレーションが行われており，トルク $\tau_2(t)$ は値が限られているにも関わらず，振り上げが成功している．

以上をまとめると $\tau_2$ を入力としてシミュレーションを行ったことにより，$\theta_2$ と $\dot{\theta}_2$ の 2 次元の状態空間表現に配置した Gauss 関数の線形和で価値関数が近似でき，制御目的を達成できている．

表 2　設定するパラメータ

| Parameter | Name | Value |
|---|---|---|
| $R(x)$ | 状態による報酬関数 | $\cos\theta_2$ |
| $c$ | 費用関数の重み | 0.1 |
| $\alpha$ | 学習率 | 1.0 |
| $\tau$ | 割引報酬の時定数 | 1.0 |
| $\kappa$ | eligbility trace の時定数 | 0.2 |
| $s_k$ | Gauss 関数の分散 | 1.0 |

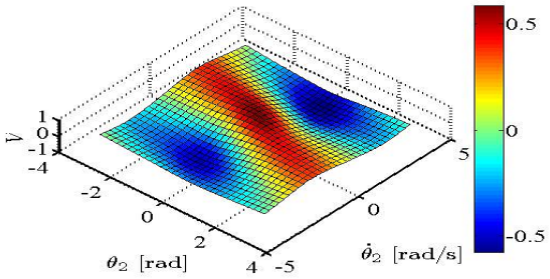


図 3　学習後の価値関数

### 4.2　回転型倒立振子の安定化問題（状態空間 3 次元）

次にリンク 1 も考え，3 次元の状態変数

$$x(t) = \left[\begin{array}{ccc} \dot{\theta}_1(t) & \theta_2(t) & \dot{\theta}_2(t) \end{array}\right]^{\mathrm{T}}$$

を持つ回転型倒立振子に強化学習を適用し，学習が成功するかどうかを検証する．制御対象の振り上げ問題を行う際には，格子点やパラメータ，サンプル時間の選び方により局所的な解にとどまることや学習が不安定になることがあり，倒立状態の近くに初期値を選択し，リンク 2 がある程度振りあがった状態から安定化するように学習をする．具体的にはリンク 2 の角度 $|\theta_2| < 0.2$[rad] の範囲で初期値をランダムに設定する．そのとき，設定したパラメータを表 3 に表す．$R(x)$ は角速度 $\dot{\theta}_2$ を含めることにする．基底関数は各軸を 16 分割して格子を作り，できた格子点をそれぞれ中心として $16^3$ 個の基底関数を用意する．$[-4, 4] \times [-\pi, \pi] \times \left[-\frac{5}{4}\pi, \frac{5}{4}\pi\right]$ の範囲にそれぞれ 16 個の基底関数を配置する．5[s] の間，$|\theta_2| < \frac{\pi}{2}$[rad] であるとき安定化を成功とする．また $|\theta_2| > \frac{\pi}{2}$ となったときは罰として，$r(x(t), v(t)) = -1$ を 1 秒間与える．5 回成功したときの試行回数は 740 回であった．次元が一つ増えることで，学習パラメータの数は 16 倍になるため，格子点の配置を工夫する必要がある．図 5 の価値関数は $\dot{\theta}_1$ 軸に垂直な平面上における，5 回安定化成功時の価値関数を示している．4.1 節でのシミュレーションに比べて 3 次元の状態空間表現を持つ回転型倒立振子の安定化には多くの試行回数を必要とすることが分かった．

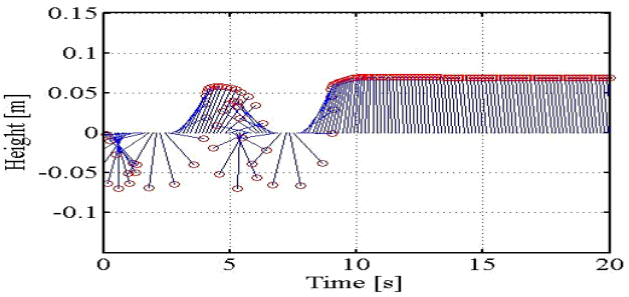


図 4　100 試行後のリンク 2 の動き

表 3　設定するパラメータ

| Parameter | Name | Value |
|---|---|---|
| $R(x)$ | 状態による報酬関数 | $\cos\theta_2 - \frac{1}{4}\dot{\theta}_2^2$ |
| $c$ | 費用関数の重み | 0.1 |
| $\alpha$ | 学習率 | 1.0 |
| $\tau$ | 割引報酬の時定数 | 1.0 |
| $\kappa$ | eligbility trace の時定数 | 0.2 |
| $s_k$ | Gauss 関数の分散 | 0.5 |

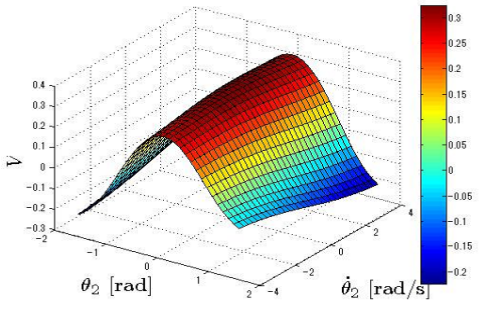


図 5　学習後の価値関数

### 4.3 INGnet による回転型倒立振子の振り上げ制御

4.2 節では，3 次元の状態空間中に等間隔に基底関数を配置した．しかし，3 次元の空間に基底関数を適切な数だけ配置することは，容易ではなく，学習時間が大幅に増加することが分かった．そこで，価値関数の近似誤差が大きく，他の基底関数と近くない状態空間に基底関数を配置する INGnet を用いる [5]．この手法は，近似誤差がある基準 $e_{\max}$ より大きく，すべての Gauss 関数があるしきい値 $a_{\min}$ より小さいとき，新しい基底関数を配置する．すなわち，新しい基底関数を配置するための条件は

$$|\delta(t)| > e_{\max} \text{ かつ } \max_k a_k < a_{\min} \qquad (16)$$

である．効率的な基底関数の配置により，振り上げはうまくいくが安定化が成功しにくいことが分かった．そこである程度振り上げ，倒立状態で報酬関数の切り替えることを考える．具体的には，

$$R(x) = \begin{cases} \cos\theta_2 & (\cos\theta_2 \leq 0.8 \text{ のとき}), \\ \cos\theta_2 - \frac{1}{4}\dot{\theta}_2^2 & (\cos\theta_2 > 0.8 \text{ のとき}) \end{cases}$$

とし，$S(v)$ は 0 とする．その理由として，振り上げと安定化では報酬関数が同一では効率的な学習が行われないと考えたためである．

表 4 にシミュレーションで設定したパラメータを示す．また，図 6 は，試行 500 回での価値関数のグラフである．図 7 は，試行 500 回後の回転型倒立振子のリンク 2 の動きを表している．図 7 から 8[s]–10[s] の間倒立状態を維持しているが，その後不安定になる．シミュレーションを行う中で，振り上げはある程度うまくいくことが確認できた．INGnet により基底関数の配置が振り上げに必要な空間に配置できていると言える．しかし，振り上がってから，安定化するためには学習時間がかかる．その理由として，倒立状態での学習が不足していることと安定化するための状態空間に適切な数だけ基底関数を配置できていないからと考えられる．

表 4 設定するパラメータ

| Parameter | Name | Value |
|---|---|---|
| $\alpha$ | 学習率 | 2.0 |
| $\tau$ | 割引報酬の時定数 | 1.0 |
| $\kappa$ | eligbility trace の時定数 | 0.2 |
| $c$ | 費用関数の重み | 0.01 |
| $s_k$ | Gauss 関数の分散 | 1.0 |
| $e_{\max}$ | $\delta(t)$ のしきい値 | 0.5 |
| $a_{\min}$ | $a_k$ のしきい値 | 0.6 |

## 5 おわりに

回転型倒立振子の振り上げ問題に対して，連続時間，連続空間の強化学習を適用した．2 次元では学習が効率的に行われ，振り上げ，安定化ができている．強化学習は次元が増えるに従い，設定するパラメータの組み合わせが容易ではなくなり，学習が不安定になることがあった．

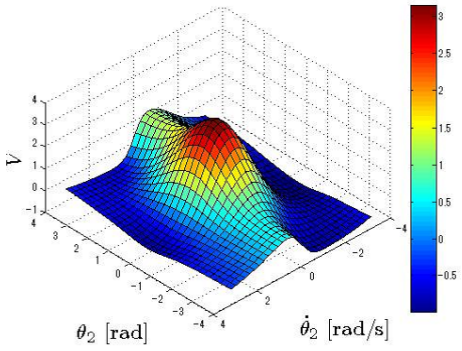


図 6 学習後の価値関数

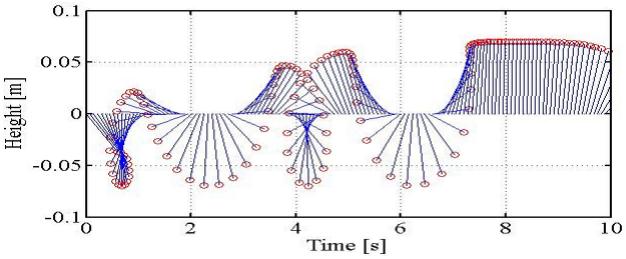


図 7 500 試行後のリンク 2 の動き

そこで，INGnet を用いてガウス関数を配置し，学習の効率を上げた．また，振り上げ問題と安定化問題で報酬関数の切り替えを行うことにより安定化するときの制御性能の改善を行った．今後の課題として，次元の大きい状態空間に配置する基底関数を適切に配置し，学習性能を向上させることである．

## 参考文献


[1] R. S. Sutton, A. G. Barto（三上貞芳，皆川雅章 訳）:『強化学習』．森北出版，東京，2000.

[2] J. Morimoto, K. Doya: Acquisition of stand-up behavior by a real robot using hierachical reinforcement learning, Robotics and Autonomous System, Vol. 36, No. 1, 37/51, 2001.

[3] 西村政哉，吉本潤一郎，時田陽一，中村奏，石井信：複数制御器の切り替え学習法による実アクロボットの制御，電子情報通信学会論文誌，Vol. J88–A，No. 5，646/657，2005.

[4] K. Doya: Reinforcement learning in continuous time and space, Neural Computation, Vol. 12, No. 1, 219/245, 2000.

[5] 森本淳，銅谷賢治：強化学習を用いた高次元連続状態における系列運動学習：起き上がり運動の獲得，電子情報通信学会論文誌，Vol. J82–D–II，No. 11，2118/2131，1999.

[6] 川田昌克：『MATLAB/Simulink と実機で学ぶ制御工学』．TechShare，東京，2013.